

### **AMENDMENTS TO THE CLAIMS**

The following listing of claims replaces all prior versions, and listings, of claims in the captioned patent application:

#### ***Listing of Claims:***

1. (Currently Amended) A method of monitoring response time of a method or a function associated with a software component, comprising the steps of:

operating on a bytecode representation of a method or function to be instrumented by inserting an instrumentation code in a ~~bytecode~~ the bytecode representation of said method or function without modifying the respective source code of said method or function and while classes of said method or function are being loaded for execution;

generating to generate a call to an Application Response Measurement (ARM) agent to cause the ARM agent to effect generation of a start time marker upon start of execution of said method or function and a stop time marker upon completion of execution of said method or function, wherein the ARM agent is one of a plurality of agents of an ARM protocol; and  
utilizing said start and stop time markers to determine a response time of said method or function.

2. (Canceled)

3. (Currently Amended) The method of ~~claim 2~~ claim 1, further comprising:  
registering said method or function with said ARM agent prior to generation of said start and stop time markers.

4. (Currently Amended) The method of ~~claim 2~~ claim 1, wherein said instrumentation code causes generation of said start and stop time markers without modifying instructions associated with execution of said method or function.

5. (Currently Amended) The method of ~~claim 2~~ claim 1, wherein said ARM agent generates a record corresponding to said method or function for storing the response time associated with said method or function.

6. (Original) The method of claim 5, wherein said record includes a field for identifying a parent, if any, of said method or function in a hierarchical parent-child transaction chain.

7. (Original) The method of claim 6, wherein said record includes another field for identifying a top level transaction in said parent-child transaction chain.

8. (Previously Presented) The method of claim 1, wherein said software component can be any of a server page, a servlet of a server side component, a driver, a naming and directory interface (NDI) or remote method invocation (RMI) component.

9. (Previously Presented) The method of claim 8, wherein said method or function of the software component comprises any of a Service method of a server page, a doFilter, a doGet, a doPost or a service method of a servlet, a getConnection, executeQuery, or selected methods of driver, or remote, local or home interface methods of a server side component.

10. (Previously Presented) The method of claim 1, wherein the step of inserting the instrumentation code comprises incorporating instrumentation hooks into said bytecode representation prior to or during loading and initialization of a class containing said method or function by a virtual machine.

11. (Original) The method of claim 1, further comprising:  
storing said response time in a database.

12. (Original) The method of claim 1, further comprising:  
displaying said response time to a user.

13. (Currently Amended) A system comprising a processor configured to monitor a response time of a method or function associated with a software component, said processor configured to implement at least:

an instrumentation engine for operating on a bytecode representation of a method or function to be instrumented by inserting instrumentation code in a ~~bytecode~~ the bytecode representation of said method or function without modifying the respective source code of said method or function and while classes of said method or function are being loaded for execution, said instrumentation code effecting generation of a start time marker and a stop time marker upon resumption and completion, respectively, of said method or function,

an interface module being invoked by said instrumentation code upon start and completion of said method or function,

an application response measurement (ARM) agent in communication with said interface module,

wherein said interface module, upon invocation by said instrumentation code, calls said ARM agent to cause generation of said start and stop time markers by said ARM agent, and wherein the ARM agent is one of a plurality of agents of an ARM protocol; and

an analysis and presentation module in communication with said ARM agent for presenting said response time to a user and/or storing said response time in a database.

14. (Previously Presented) The system of claim 13, wherein said instrumentation engine inserts said instrumentation code prior to loading of a class containing said method or function by a virtual machine.

15. (Original) The system of claim 13, wherein said instrumentation engine inserts said instrumentation code in said bytecode representation without modifying instructions associated with execution of said method or function.

Application No: 10/640,623  
Atty. Docket: 10017138-1

Response to Action  
dated July 17, 2007

16-21 (Canceled).